APPEAL BRIEF

Katherine R. Vieyra
Attorney for Appellants
Registration No. 47,155

SCULLY, SCOTT, MURPHY & PRESSER, P.C.
400 Garden City Plaza, Suite 300
Garden City, New York 11530
(516) 742-4343

TABLE OF CONTENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Hiroshi CHISHIMA	**Docket:**	17261
Serial No.:	10/717,867	**Examiner:**	Matthew J. Ludwig
Filed:	November 19, 2003	**Art Unit:**	2178
For:	FUNCTION EXTENSION TYPE BROWSER, BROWSER COMPONENT PROGRAM AND RECORDING MEDIUM	**Dated:**	February 21, 2008
Confirm No:	9342		

Mail Stop Appeal Brief- Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

Pursuant to 35 U.S.C. § 134 and 37 C.F.R. § 41.37, entry of this Appeal Brief in support of the Notice of Appeal filed October 11, 2007 in the above-identified matter is respectfully requested. This paper is submitted as a brief setting forth the authorities and arguments upon which Appellant relies in support of the appeal from the August 9, 2007 Final Rejection of Claims 1, 3-23, 28 and 29 in the above-identified patent application.

I. REAL PARTY IN INTEREST

The real party of interest in the above-identified patent application is NEC Corporation.

II. RELATED APPEALS AND INTERFERENCES

There are no pending appeals or interferences related to this application to Appellant's knowledge.

III. STATUS OF CLAIMS

Claim 1 stands rejected based on 35 U.S.C. § 103(a) as unpatentable over Shigemi et al., U.S. Patent No. 6,314,434 (hereinafter "Shigemi").

Claim 2 stands canceled.

Claims 3-23 stand rejected based on 35 U.S.C. § 103(a) as unpatentable over Shigemi.

Claims 24-27 stand canceled.

Claims 28-29 stand rejected based on 35 U.S.C. § 103(a) as unpatentable over Shigemi.

Claims 1, 3-23, and 28-29 are appealed; a clean copy of these claims is attached hereto in the Claims Appendix.

IV. STATUS OF AMENDMENTS

No amendment was filed in Response to the Final Rejection mailed August 9, 2007.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Claims 1, 3-23, 28 and 29 are the claims on appeal. A copy of the rejected claims is attached hereto in the Claims Appendix.

The invention with respect to independent Claim 1 comprises a function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document parser unit (#23) for converting document data into structured document information (page 14, line 25 to page 15, line 2) in response to an instruction from the application program (#4, #4a, page 20, lines 16-20); a document information manipulation unit (#24) for enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8); a browser core unit (#22) for displaying a display document based on the structured document information in response to an instruction from the application program (#4, #4a, page 22, lines 1-6); and an event information informing unit (#25), wherein when an event relating to the display document takes place, said event informing unit (#25) informs the application program (#4, #4a) of event information indicating a type of the event and a part of the display document where the event takes place (page 15, lines 11-17, page 22, lines 15-21).

The invention with respect to independent Claim 18 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document information manipulation unit (#24) for enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8); a browser core unit (#22) for displaying a document based on the structured document information according to an instruction from the application program (#4, #4a, page 22, lines 1-6); and an event information informing unit (#25) for, when an event relating to the displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 22, lines 15-21); wherein the browser component converts document data into structured document information in response to an instruction from the application program (#4, #4a, page 20, lines 16-22).

The invention with respect to independent Claim 19 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document parser unit (#23) for converting document data into structured document information (page 14, line 25 to page 15, line 2) according to an instruction from the application program (#4, #4a, page 20, lines 16-20); a browser core unit (#22) for displaying a document based on the structured document information according to an instruction from the application program (#4, #4a, page 22, lines 1-6); and an event information informing unit (#25) for, when an event relating to the displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 15, lines 11-23); wherein the browser component enables the structured document information to be referred to from the application program (#4, #4a, page 20, lines 16-22).

The invention with respect to independent Claim 20 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document parser unit (#23) for converting document data into structured document information (page 14, line 25 to page 15, line 2) according to an instruction from the application program (#4, #4a, page 20, lines 16-20); a document information manipulation unit (#24) for enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8); and a browser core unit (#22) for displaying a document based on the structured document information according to an instruction from the application program (#4, #4a, page 22, lines 1-6); wherein the browser component, when an event relating to the displayed document takes place, informs the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 15, lines 11-23).

The invention with respect to independent claim 21 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, 4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document information manipulation unit (#24) for enabling structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8) and editing the structured document information according to an instruction from the application program (#4, #4a, page 21, lines 20-22); a browser core unit (#22) for displaying a document based on the structured document information edited in the document information manipulation unit (#24) according to an instruction from the application program (#4, #4a, page 22, lines 1-6); and an event information informing unit (#25) for, when an event relating to a displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 22, lines 15-21); wherein the browser component converts document data into structured document information in response to an instruction from the application program (#4, #4a, page 20, lines 16-22).

The invention with respect to independent Claim 22 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); a document parser unit (#23) for converting document data into structured document information (page 14, line 25 to page 15, line 2) according to an instruction from the application program (#4, #4a, page 20, lines 16-20); a browser core unit (#22) for displaying a document based on the structured document information edited according to an instruction from the application program (#4, #4a, page 22, lines 1-6); and an event information informing unit (#25) for, when an event relating to a displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 15, lines 11-17); wherein the browser component enables the structured document information to be referred to from the application program (#4, #4a, page 15, line 24 to page 16, line 22) and edits the structured document information in response to an instruction from the application program (#4, #4a, page 21, lines 17-19).

The invention with respect to independent Claim 23 comprises a browser component being components of a function extension type browser (#2), wherein the function extension type browser (#2) comprising an application program (#4, #4a) downloaded when an information service requiring an extension of a markup language or meta-information is used (#4, #4a, page 17, lines 5-8); a document parser unit (#23) for converting document data into structured document information (page 14, line 25 to page 15, line 2) according to an instruction from the application program (#4, #4a, page 20, lines 16-20); a document information manipulation unit (#24) for enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8) and editing the structured document information according to an instruction from the application program (#4, #4a, page 21, lines 17-19); and a browser core unit (#22) for displaying a document based on the structured document information edited in the document information manipulation unit (#24) according to an instruction from the application program (#4, #4a, page 22, lines 1-6); wherein the browser component, when an event relating to a displayed document takes place, informs the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 15, lines 11-17).

The invention with respect to independent claim 28 comprises a program storage device, tangibly embodying a program of instructions executable by a machine for performing a method comprising the steps of converting document data into structured document information (Page 14, line 25 to page 15, line 2) according to an instruction from an application program (#4, #4a, page 20, lines 16-20) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8); displaying a document based on the structured document information according to an instruction from the application program (#4, #4a, page 22, lines 1-6); and when an event relating to a displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the event takes place (page 15, lines 11-17, page 22, lines 15-21).

The invention with respect to independent claim 29 comprising a program storage device, tangibly embodying a program of instructions executable by a machine for performing a method comprising the steps of converting document data into structured document information (page 14, line 25 to page 15, line 2) according to an instruction from an application program (#4, #4a, page 20, lines 16-20) downloaded when an information service requiring an extension of a markup language or meta-information is used (page 17, lines 5-8); enabling the structured document information to be referred to from the application program (#4, #4a, page 15, lines 4-8) and editing the structured document information according to an instruction from the application program (#4, #4a, page 21, lines 12-19); displaying a document based on the structured document information according to another instruction from the application program (#4, #4a, page 22, lines 1-6); and when an event relating to a displayed document takes place, informing the application program (#4, #4a) of event information indicating a type of the event and a part of the document where the even takes place (page 15, lines 11-17, page 22, lines 15-21).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 3-23, 28 and 29 are rejected under 35 U.S.C. § 103(a) as unpatentable over Shigemi.

VII. ARGUMENT

(A) Examiner's Rejection Of Independent Claims 1, 20, 22, 23, 28 and 29 Is Not Proper.

(I) Reference Does Not Teach or Suggest a parsing technique utilized with SGML documents.

Appellant submits that Shigemi does not disclose or suggest parsing or a document parser unit for converting document data into structured document information, and Shigemi does not provide a proficient suggestion of a parsing technique utilized with SGML documents as recited in independent claims 1, 20, 22, 23, 28 and 29.

The Examiner acknowledges that Shigema does not explicitly state a document parser, but contends that Shigema provides a suggestion of parsing SGML and XML data into structured document data (and/or nodes). Shigemi discloses the following operation. A script interpreter receives event messages from external sources, such as a user, and manipulates these incoming messages. Next, DTD, e.g. structured documents, are read and a relationship description is associated with the DTD, from which an operator to be executed is determined. Finally, the script of the operator is executed. (see column 14, line 60 to column 15, line 29).

Shigemi does not disclose or suggest converting document data into structured document information. Instead, Shigemi discloses only that already structured data, such as SGML or XML, is merely transported (not converted), as taught, for example, in column 9, lines

24-36. Shigema teaches that the message delivery subsystem is a mechanism to transfer events (column 9, lines 42-43). Further, the script interpreter disclosed by Shigema parses and executes MIPS scripts (column 10, lines 27-35); it does not convert data into structured data. Hence, Shigema does not disclose or suggest either parsing or a document parser unit for *converting document data into structured document information* as recited in independent claims 1, 20, 22, 23, 28, and 29.

It has been held by the courts that to establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. See, *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). Further, the skill in the art component will rarely operate to supply missing knowledge or prior art to reach an obviousness judgment. *Al-Site Corp. v. VSI International Inc.*, 174 F.3d 1308, 50 USPQ2d 1161 (Fed. Cir. 1999). As discussed above, Shigema does not teach or suggest all features of independent claims 1, 20, 22, 23, 28 and 29. Knowledge of one skilled in the art does not overcome this deficiency. As discussed above, appellant submits that one skilled in the art would not read Shigema as teaching or suggesting the limitation of converting document data into structured document information. Hence, *prima facie* obviousness has not been evidenced. Accordingly, appellant respectfully requests that the Board withdraw the rejection of independent claims 1, 20, 22, 23, 28 and 29.

(II) Reference Does Not Teach or Suggest an application program that enables structured document information to be referred to by a browser.

The Examiner also alleges that the scripts of Shigema read on an application program and refer to an extension of markup language or meta information. Appellant respectfully disagrees. The application program recited in the claims of the present invention is a program that enables structured document information to be referred to by a browser, or to be converted from document data into structured information.

In contrast, while Shigema discloses that XML can be used as an alternative to SGML, Shigema does not disclose or suggest that XML can be used as an alternative to scripts (column 9, lines 19-22). In fact, Shigema does not disclose or suggest any relationship between scripts and XML. Thus, even assuming, *inter alia*, that the scripts of Shigema were an application program, Shigema does not disclose or suggest an application program downloaded when an information service requiring an extension of a markup language or meta-information is used, so that the application program enables the reference to structured document information or the conversion of document data into structured information as recited in the independent claims.

Thus, Shigema fails to disclose or suggest a downloaded application program enabling manipulation of structured document information, and the Examiner fails to evidence *prima facie* obviousness. Accordingly, appellant respectfully requests that the Board withdraw the rejection of independent claims 1, 18, 19, 20, 21, 22, 23, 28 and 29.

(B) Rejection of Dependent Claims is Also Improper.

(I) Claims 3-17 are Patentable Based Upon Dependency From Independent Claim 1.

Appellant respectfully submits that Claims 3-17 are patentable over the cited prior art based upon at least the analysis provided above. Specifically, Shigema fails to teach, suggest, or render obvious each and every limitation of independent Claim 1, from which Claims 3-17 depend.

Withdrawal of the rejection to the dependent Claims 3-17 is, therefore, respectfully requested.

(C) Conclusion

Based on the above arguments and remarks, Appellant respectfully submits that the claims of the instant invention on appeal are not anticipated or obvious in light of Shigema. Consequently, the rejection of the claims based on this reference is in error. In view of the remarks submitted hereinabove, the reference applied against Claims 1, 3-23, 28 and 29 on appeal does not render those claims unpatentable under 35 U.S.C. § 103(a). Thus, Appellant submits that the § 103 rejection is in error and must be reversed.

The Commissioner is hereby authorized to charge any additional fees or credit any overpayment in connection herewith to Deposit Account No. 19-1013/SSMP.

Respectfully submitted,

Katherine R. Vieyra

Katherine R. Vieyra
Registration No. 47,155

SCULLY SCOTT MURPHY & PRESSER, P.C.
400 Garden City Plaza, Suite 300
Garden City, New York 11530
(516) 742-4343

VIII. CLAIMS APPENDIX

1. **(Rejected)** A function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document parser unit for converting document data into structured document information in response to an instruction from the application program;

a document information manipulation unit for enabling the structured document information to be referred to from the application program;

a browser core unit for displaying a display document based on the structured document information in response to an instruction from the application program; and

an event information informing unit, wherein:

when an event relating to the display document takes place, said event informing unit informs the application program of event information indicating a type of the event and a part of the display document where the event takes place.

2. (Canceled)

3. (Rejected) The function extension type browser as claimed in claim 1, wherein the document information manipulation unit has a function of editing the structured document information in response to an instruction from the application program.

4. **(Rejected)** The function extension type browser as claimed in claim 1, wherein the application program has a function of, when the event information is informed from the event information informing unit, referring to the structured document information by using the

document information manipulation unit, obtaining a content of a node relating to the part where the event takes place among nodes being components of the structured document information, and performing processing according to the type of the event indicated by the informed event information and the obtained content of the node.

5. **(Rejected)** The function extension type browser as claimed in claim 1, wherein the application program has a function of, when the event information is informed from the event information informing unit, referring to the structured document information by using the document information manipulation unit, obtaining a content of a node relating to the part indicated by the event information among nodes being components of the structured document information, and when the obtained content of the node relates to a proprietary tag or a proprietary attribute, performing processing in accordance with the obtained content of the node and the type of the event indicated by the event information.

6. **(Rejected)** The function extension type browser as claimed in claim 5, wherein the processing performed in accordance with the obtained content of the node and the type of the event indicated by the event information is voice production processing.

7. **(Rejected)** The function extension type browser as claimed in claim 3, wherein the application program has:

a function of instructing the document information manipulation unit regarding a node intended to be edited, said node from among nodes being components of the structured document information, and instructing the document information manipulation unit regarding a content of the editing, and

a function of, when the event information informing unit notifies said application program of the type of the event, referring to the structured document information using the document information manipulation unit, obtaining a content of said node relating to the part indicated by the event information informing unit and performing processing according to the type of the event indicated by the event information informing unit and the obtained content of the node.

8. (Rejected) The function extension type browser as claimed in claim 3, wherein the application program includes:

a function of instructing the document information manipulation unit regarding a node intended to be edited, said node from among nodes being components of the structured document information, and instructing the document information manipulation unit regarding a content of the editing, and

a function of, when the event information is informed from the event information informing unit, referring to the structured document information using the document information manipulation unit, obtaining a content of a node relating to the part indicated by the event information, said node from among the nodes being the components of the structured document information, and if the obtained content of the node is a proprietary tag or a proprietary attribute which are uniquely extended attributes not existent in a current markup language, performing processing according to the obtained content of the node and the type of the event indicated by the event information.

9. (Rejected) The function extension type browser as claimed in claim 8, wherein the event information informing unit has a function of, when an event relating to a document to be displayed takes place, informing the application program of event information indicating the type of the event, and

the application program has a function of, when the event information is informed from the event information informing unit, instructing the document information manipulation unit regarding the node intended to be edited, said node from among nodes being components of the structured document information corresponding to the document to be displayed regarding the content of the editing.

10. (Rejected) The function extension type browser as claimed in claim 8, wherein

the node intended to be edited is a node corresponding to the proprietary tag or the proprietary attribute which are uniquely extended attributes not existent in a current markup language, the content of the editing is to cause the node corresponding to the proprietary tag or the proprietary attribute to be a comment node, and the processing performed according to the proprietary tag or the proprietary attribute is voice production processing.

11. (Rejected) The function extension type browser as claimed in claim 9, wherein the node intended to be edited is a node corresponding to the proprietary tag or the proprietary attribute which are uniquely extended attributes not existent in a current markup language, the content of the editing is to cause the node corresponding to the proprietary tag or the proprietary attribute to be a comment node, and the processing performed according to the proprietary tag or the proprietary attribute is voice production processing.

12. (Rejected) The function extension type browser as claimed in claim 1, wherein the application program has a function of, when the event information is informed from the event information informing unit, referring to the structured document information by using the document information manipulation unit, obtaining a content of a node corresponding to the part indicated by the event information among nodes being components of the structured document information, and based on the obtained content, creating meta-information in a format not supported by the function extension type browser.

13. **(Rejected)** The function extension type browser as claimed in claim 1, wherein the application program has:

a function of, according to the event information informed from the event information informing unit, creating an HTTP request header in a format not supported by the browser core unit and issuing an HTTP request including the created HTTP request header, and a function of obtaining document data and performing processing

relating to an HTTP response header accompanying the document data.

14. **(Rejected)** The function extension type browser as claimed in claim 13, wherein the HTTP response header and the HTTP request header handled by the application program are an HTTP response header and an HTTP request header relating to a cookie.

15. **(Rejected)** The function extension type browser as claimed in claim 1, wherein the application program has a function of analyzing document data in a format not supported by the document parser unit, and based on an analysis result, creating structured document information by using the document information manipulation unit.

16. **(Rejected)** The function extension type browser as claimed in claim 1, comprising a memory for storing the application program.

17. **(Rejected)** The function extension type browser as claimed in claim 3, wherein the structured document information which is used by the browser to display the document is edited at the document information manipulation unit.

18. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document information manipulation unit for enabling the structured document information to be referred to from the application program;

a browser core unit for displaying a document based on the structured document information according to an instruction from the application program; and

an event information informing unit for, when an event relating to the displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the event takes place; wherein:

the browser component converts document data into structured document information in response to an instruction from the application program.

19. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document parser unit for converting document data into structured document information according to an instruction from the application program;

a browser core unit for displaying a document based on the structured document information according to an instruction from the application program; and

an event information informing unit for, when an event relating to the displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the event takes place; wherein:

the browser component enables the structured document information to be referred to from the application program.

20. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document parser unit for converting document data into structured document information according to an instruction from the application program;

a document information manipulation unit for enabling the structured document information to be referred to from the application program; and

a browser core unit for displaying a document based on the structured document information according to an instruction from the application program; wherein:

the browser component, when an event relating to the displayed document takes place, informs the application program of event information indicating a type of the event and a part of the document where the event takes place.

21. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document information manipulation unit for enabling structured document information to be referred to from the application program and editing the structured document information according to an instruction from the application program;

a browser core unit for displaying a document based on the structured document information edited in the document information manipulation unit according to an instruction from the application program; and

an event information informing unit for, when an event relating to a displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the event takes place; wherein:

the browser component converts document data into structured document information in response to an instruction from the application program.

22. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document parser unit for converting document data into structured document information according to an instruction from the application program;

a browser core unit for displaying a document based on the structured document information edited according to an instruction from the application program; and

an event information informing unit for, when an event relating to a displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the event takes place; wherein:

the browser component enables the structured document information to be referred to from the application program and edits the structured document information in response to an instruction from the application program.

23. **(Rejected)** A browser component being components of a function extension type browser, wherein the function extension type browser comprising:

an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

a document parser unit for converting document data into structured document information according to an instruction from the application program;

a document information manipulation unit for enabling the structured document information to be referred to from the application program and editing the structured document information according to an instruction from the application program; and

a browser core unit for displaying a document based on the structured document information edited in the document information manipulation unit according to an instruction from the application program; wherein

the browser component, when an event relating to a displayed document takes place, informs the application program of event information indicating a type of the event and a part of the document where the event takes place.

24. – 27. **(Cancelled)**

28. **(Rejected)** A program storage device, tangibly embodying a program of instructions executable by a machine for performing a method comprising the steps of:

converting document data into structured document information according to an instruction from an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

enabling the structured document information to be referred to from the application program;

displaying a document based on the structured document information according to an instruction from the application program; and

when an event relating to a displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the event takes place.

29. **(Rejected)** A program storage device, tangibly embodying a program of instructions executable by a machine for performing a method comprising the steps of:

converting document data into structured document information according to an instruction from an application program downloaded when an information service requiring an extension of a markup language or meta-information is used;

enabling the structured document information to be referred to from the application program and editing the structured document information according to an instruction from the application program;

displaying a document based on the structured document information according to another instruction from the application program; and

when an event relating to a displayed document takes place, informing the application program of event information indicating a type of the event and a part of the document where the even takes place.

IX. EVIDENCE APPENDIX

None.

X. RELATED PROCEEDINGS APPENDIX

None. There are no related proceedings.